ELSEVIER

Journal of Computational and Applied Mathematics 125 (2000) 171–182

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

Numerical solutions of stochastic differential equations – implementation and stability issues

Kevin Burrage[a,*], Pamela Burrage[a], Taketomo Mitsui[b]

[a] *Department of Mathematics, University of Queensland, St Lucia, Queensland, 4072, Australia*

[b] *Graduate School of Human Informatics, Nagoya University, Furo-cho Chikusa-ku, Nagoya 464-8601, Japan*

Received 25 October 1999

Abstract

Stochastic differential equations (SDEs) arise from physical systems where the parameters describing the system can only be estimated or are subject to noise. There has been much work done recently on developing numerical methods for solving SDEs. This paper will focus on stability issues and variable stepsize implementation techniques for numerically solving SDEs effectively.

Keywords: Stability; Stochastic differential equations; Variable stepsize

1. Introduction

This paper presents an overview of stability and implementation issues of numerical methods for solving stochastic differential equations. Due to space constraints it is not possible to give details behind the construction of numerical methods suitable for solving SDEs, instead the paper will focus on the stability and implementation of numerical methods. Thus Section 2 discusses numerical stability both of SDEs and of numerical methods for solving these SDEs, while the implementation of numerical methods using a fixed stepsize is discussed in Section 3; in Section 4 a variable stepsize implementation is presented.

This section continues with some necessary background details covering the form of an SDE together with definitions of order of convergence for numerical methods to solve such SDEs.

Stochastic differential equations describe physical systems where noise is present, with the noise being modelled by a Wiener process that is nowhere differentiable. The general form of an autonomous

* Corresponding author.

E-mail address: kb@maths.uq.edu.au (K. Burrage).

0377-0427/00/$ - see front matter
PII: S0377-0427(00)00467-2

SDE is

$$\mathrm{d}y(t) = f(y(t))\,\mathrm{d}t + g(y(t))\,\mathrm{d}W(t), \quad y \in \mathbb{R}^m, \quad y(0) = y_0, \tag{1}$$

where f is the drift coefficient (an m-vector-valued function), g is the diffusion coefficient (an $m \times d$ matrix-valued function), and $W(t)$ is a d-dimensional process having independent scalar Wiener process components ($t \geqslant 0$). A Wiener process W is a Gaussian process with the property that

$$\boldsymbol{E}(W(t)) = 0, \quad \boldsymbol{E}(W(t)W(s)) = \min\{t, s\}.$$

The Wiener increments $W(t) - W(s)$ are independent Gaussian processes with mean 0 and variance $|t - s|$.

Eq. (1) can be written as a stochastic integral equation

$$y(t) = y(t_0) + \int_{t_0}^{t} f(y(s))\,\mathrm{d}s + \int_{t_0}^{t} g(y(s))\,\mathrm{d}W(s)$$

where the first integral is a regular Riemann–Stieltjes integral and the second integral is a stochastic integral, commonly interpreted in either Itô or Stratonovich form. The Stratonovich interpretation follows the usual rules of Riemann–Stieltjes calculus, and for this reason is the form used in this paper (the symbol $\circ$ in front of $\mathrm{d}W(s)$ will serve to confirm a Stratonovich integral). However, an SDE presented in Itô form can be converted to Stratonovich form using a simple formula which relates the two interpretations. Indeed the solution of (1) and its related Stratonovich SDE are exactly the same:

$$\mathrm{d}y(t) = \bar{f}(y(t)) + g(y(t)) \circ \mathrm{d}W(t), \tag{2}$$

$$\bar{f}_i(y(t)) = f_i(y(t)) - \tfrac{1}{2}\sum_{j=1}^{m}\sum_{k=1}^{d} g_{jk}(y(t))\frac{\partial g_{ik}(y(t))}{\partial y_j}, \quad i = 1, \ldots, m. \tag{3}$$

A multiple Stratonovich integral is given by

$$J_{j_1 j_2 \cdots j_l}(t_0, t) = \int_{t_0}^{t}\int_{t_0}^{s_l} \cdots \int_{t_0}^{s_2} \circ \mathrm{d}W_{s_1}^{j_1} \circ \cdots \circ \mathrm{d}W_{s_l}^{j_l},$$

where $j_l \in \{0, 1, \ldots, d\}$ for d Wiener processes. Note that the integral $J_0(t_0, t) = \int_{t_0}^{t} \circ \mathrm{d}W_{s_1}^0 = \int_{t_0}^{t} \mathrm{d}s_1$. For ease of notation, the written dependence on t_0 and t will be dropped when the meaning is clear from the context.

There are two ways of measuring the accuracy of a numerical solution of an SDE – these are strong convergence and weak convergence – only strong convergence will be considered in this paper. Strong convergence is required when each trajectory of the numerical solution must be close to the exact solution:

Definition 1. Let $\bar{y}_N$ be the numerical approximation to $y(t_N)$ after N steps with constant stepsize $h = (t_N - t_0/N)$; then $\bar{y}$ is said to *converge strongly* to y with strong global order p if $\exists C > 0$ (independent of h) and $\delta > 0$ such that

$$\boldsymbol{E}(\|\bar{y}_N - y(t_N)\|) \leqslant Ch^p, \quad h \in (0, \delta).$$

This definition is for global order – the local error can behave as $O(h^{p+1/2})$; fractional orders arise as the root mean square order of the Wiener process is $h^{1/2}$.

Numerical methods for SDEs are derived by comparing the stochastic Taylor series expansion of the numerical solution with that of the exact solution, over one step assuming exact initial values. This comparison results in a set of order conditions to be satisfied – see [2,13] for the development of these order conditions using Rooted Tree theory in the case of Stratonovich problems.

This section has provided an overview of the basic definitions required for studying numerical methods for solving SDEs; in the next section, stability of the SDE and of numerical methods is discussed.

2. Numerical stability analysis

As in other areas of numerical analysis, *numerical stability* is significant in the case of SDEs which usually require a long (numerical) time-integration.

2.1. Stochastic stability

Consider the scalar version of (1). We assume that there exists a unique solution $y(t; t_0, y_0)$ of the equation for $t > t_0$. Moreover, we suppose that the equation allows a steady solution $y(t) \equiv 0$. This means that $f(0) = g(0) = 0$ holds. A steady solution is often called an *equilibrium position.*

Has'minskii [10] gave the following three definitions of stability.

Definition 2. The equilibrium position of the SDE is said to be stochastically stable, stochastically asymptotically stable and stochastically asymptotically stable in the large, respectively, if the following conditions hold:

(i) For all positive ε and for all t_0 the following equality holds.

$$\lim_{y_0 \to 0} P\left(\sup_{t \geqslant t_0} |y(t; t_0, y_0)| \geqslant \varepsilon\right) = 0$$

(ii) In addition to the above,

$$\lim_{y_0 \to 0} P\left(\lim_{t \to \infty} |y(t; t_0, y_0)| = 0\right) = 1.$$

(iii) Moreover to the above two,

$$P\left(\lim_{t \to \infty} |y(t; t_0, x_0)| = 0\right) = 1 \quad \text{for all } y_0.$$

Each item in Definition 2 can be seen as the stochastic counterparts of stability, asymptotic stability and asymptotic stability in the large, respectively, in the ODE case.

Actually we can derive a criterion of the asymptotic stochastic stability for the SDE. Assume that the functions f and g are uniformly asymptotically linear with respect to x; that is, for certain real constants a and b,

$$f(x) = ax + \bar{f}(x), \quad g(x) = bx + \bar{g}(x)$$

with

$$\lim_{|x|\to 0} \frac{|\bar{f}(x)| + |\bar{g}(x)|}{|x|} = 0$$

hold uniformly in t. The solution $y(t)$ of the SDE is stochastically asymptotically stable if $a - b^2/2 < 0$. This criterion found in [8, p. 139] strongly suggests a possibility of analogous *linear stability* analysis for numerical schemes of SDE to those of ODE. We can consider that the linear parts of f and g are dominant in the asymptotic behaviour of solutions around the equilibrium position.

2.2. Numerical asymptotic stability

To cope with linear stability analysis, we introduce a linear test equation (supermartingale equation)

$$\mathrm{d}y(t) = \lambda y(t)\,\mathrm{d}t + \mu y(t)\,\mathrm{d}W \quad (t > 0) \quad \text{with } \lambda, \mu \in \mathbb{C} \tag{4}$$

with the initial condition $y(0) = 1$ to the numerical stability analysis. Since the exact solution of (4) is written as

$$y(t) = \exp\{(\lambda - \tfrac{1}{2}\mu^2)t + \mu W(t)\},$$

it is quite easy to show that the equilibrium position $y(t) \equiv 0$ is stochastically asymptotically stable if

$$\mathrm{Re}\,(\lambda - \tfrac{1}{2}\mu^2) < 0. \tag{5}$$

We can arrive at the following definition which is found in [11].

Definition 3. When a numerical scheme is applied to the stochastically asymptotically stable equation (4) and generating the sequence $\{y_n\}$, it is said to be numerically asymptotically stable if

$$\lim_{n\to\infty} |y_n| = 0 \quad \text{with probability 1.}$$

To analyze the behaviour of real stochastic processes derived from various numerical schemes, the following lemma given in [11] is useful.

Lemma 4. *Given a sequence of real-valued, nonnegative, independent and identically distributed random variable* $\{Z_n\}$, *consider the sequence random variable* $\{Y_n\}$ *defined by*

$$Y_n = \left(\prod_{i=0}^{n-1} Z_i\right) Y_0,$$

where $Y_0 \geqslant 0$ *and* $Y_0 \neq 0$ *with probability* 1. *Suppose that the random variable* $\log(Z_i)$ *are square-integrable. Then* $\lim_{n\to\infty} Y_n = 0$ *with probability* 1 *iff* $\boldsymbol{E}(\log(Z_i)) < 0$ *for all i.*

However, the numerical asymptotic stability criterion does not work well. The reason is that criterion (5) allows the case $\mathrm{Re}\,\lambda > 0$. It implies that some sample paths of the solution surely

decrease to 0, whereas their distributions possibly *increase*. This can be understood through the fact that when $\mathrm{Re}\,\lambda > 0$ the equation cannot be asymptotically stable even in the ODE sense. Henceforth, it is impossible to carry out a numerical scheme *until* all the sample paths of the exact solution diminish to 0 if two conditions $\mathrm{Re}\,\lambda > 0$ and $\mathrm{Re}\,(\lambda - \frac{1}{2}\mu^2) < 0$ are valid simultaneously. Since the numerical solution would reflect this statistical property, nobody can expect a numerically stable solution. Even in the case of the stochastic θ-method given by

$$y_{n+1} = y_n + (1-\theta)f(y_n)h + \theta f(y_{n+1})h + g(y_n)\Delta W_n \quad (\theta \in [0,1]) \tag{6}$$

there are combinations of the parameters λ and μ in (4) which do not give numerical asymptotic stability with any h (see [11]).

This investigation implies the necessity of another stability concept for SDEs. That is, we try to answer the question what SDE has all sample paths whose distribution tends to 0 as $t \to \infty$.

2.3. MS-stability

Analysis of the previous subsection suggests an introduction of a norm of the SDE solution with respect to the underlying stability concept.

Definition 5. The equilibrium position $y(t) \equiv 0$ is said to be asymptotically stable in pth mean if for all positive ε there exists a positive δ which satisfies

$$\boldsymbol{E}(|y(t)|^p) < \varepsilon \quad \text{for all } t \geqslant 0 \quad \text{and} \quad |y_0| < \delta \tag{7}$$

and, furthermore, if there exists a positive δ_0 satisfying

$$\lim_{t\to\infty} \boldsymbol{E}(|y(t)|^p) = 0 \quad \text{for all } |y_0| < \delta_0. \tag{8}$$

The most frequently used case $p = 2$ is called the *mean-square case.* Thus we introduce the norm of the solution by $\|y\| = \{\boldsymbol{E}|y|^2\}^{1/2}$.

The necessary and sufficient condition is rather simple (see [18]).

Lemma 6. *The linear test equation* (*supermartingale equation*) (4) *with the unit initial value is asymptotically stable in the mean-square sense* (*abbreviated as MS-stability*) *iff* $\mathrm{Re}\,\lambda + |\mu|^2/2 < 0$.

Note that since the inequality $\mathrm{Re}(2\lambda - \mu^2) \leqslant 2\mathrm{Re}\,\lambda + |\mu|^2$ is always valid, the asymptotic stability in the mean-square sense implies the stochastic stability.

2.4. Numerical MS-stability

For asymptotically MS-stable problems of SDEs, what conditions are imposed to derive numerically asymptotically MS-stable solutions? That is to say, what conditions should be for the numerical solution $\{y_n\}$ of the linear test equation (4) to achieve $\|y_n\| \to 0$ as $n \to \infty$?

Denote $\mathbf{E}|y_n|^2$ by Y_n. When we apply a numerical scheme to the linear test equation and take the mean-square norm, we obtain a one-step difference equation of the form $Y_{n+1} = R(\bar{h}, k)Y_n$ where two scalars $\bar{h}$ and k stand for $h\lambda$ and μ^2/λ, respectively. We can call $R(\bar{h}, k)$ the *stability function* of the scheme, and arrive at the following.

Definition 7 (Saito and Mitsui [18]). The scheme is said to be numerically MS-stable for $\bar{h}$ and k if its stability function $R(\bar{h},k)$ is less than unity in magnitude. The set in $\mathbb{C}^2$ given by $\mathscr{R} = \{(\bar{h},k);\ |R(\bar{h},k)| < 1 \text{ holds}\}$ is called the domain of MS-stability of the scheme.

In addition, we can say that a numerical scheme is A-stable if it is MS-stable for arbitrary h which is sufficiently small for the convergence.

We will derive the stability function of some numerical schemes known in the literature. Details with figures can be seen in [18].

First is the Euler–Maruyama scheme (6) (with $\theta = 0$), whose application to (4) implies

$$y_{n+1} = y_n + h\lambda y_n + \mu y_n \Delta W_n.$$

We obtain the stability function as

$$R(\bar{h},k) = |1 + \bar{h}|^2 + |k\bar{h}|.$$

Fortunately, the function depends on $\bar{h}$ and $|k|$, not on k. Therefore, we obtain the domain of MS-stability in the three-dimensional space of $(\bar{h},|k|)$.

Next is the stochastic θ-method (6). Note, we assume the implicitness only on the drift term f. A calculation leads to the stability function

$$R(\bar{h},k,\theta) = \frac{|1 + (1-\theta)\bar{h}|^2 + |k\bar{h}|}{|1 - \theta\bar{h}|^2}.$$

By comparing the regions of MS-stability of the Euler–Maruyama and the semi-implicit Euler schemes under the restriction of real $\bar{h}$ and k we can see that the latter is superior to the former with respect to the stability. Further discussion is carried out in [11].

2.5. T-stability

From the viewpoint of computer implementation, MS-stability may still cause difficulty. To evaluate the quantity of the expectation $Y_n = \mathbf{E}(|y_n|^2)$ where y_n is an approximating sequence of the solution sample path, in a certain probability y_n *happens to overflow* in computer simulations. This actually violates the evaluation of y_n.

The above situation suggests an introduction of another stability notion with respect to the approximate sequence of sample path (trajectory). It must take into account the *driving process,* whose way of realization a numerical scheme for SDE requires for the increment ΔW_n of the Wiener process. For example, in the Euler–Maruyama scheme given in (6) ΔW_n, which stands for $W(t_{n+1}) - W(t_n)$, can be exactly realized with $\xi_n\sqrt{h}$. More sophisticated schemes need more complicated normal random variables. And these random variables are to be realized through an *approximation* with pseudo-random numbers on computer, for the normal random number requires infinitely many trials. Therefore, we arrive at the following.

Definition 8. Assume that the test equation (4) is stochastically asymptotically stable in the large. The numerical scheme equipped with a specified driving process said to be T-stable if $|y_n| \to 0$ $(n \to \infty)$ holds for the driving process.

The above definition gives rise to another problem: a criterion of T-stability depends not only on the scheme but also on the driving process. It causes our analysis more difficulty. To resolve it, we can employ Lemma 4 again. For example, if the Euler–Maruyama scheme is applied to (4) then the quantity $T(h;\lambda,\mu)$ defined through

$$\log T(h;\lambda,\mu)=\int_{-\infty}^{\infty}\log|1+\lambda h+\mu\sqrt{h}x|\,p(x)\,\mathrm{d}x$$

can stand for the T-stability function of the scheme, for $T(h;\lambda,\mu)<1$ implies the T-stability.

For an illustration, we treat the Euler–Maruyama scheme with three-point random variables. The random variable ζ_n is taken as $U_n\sqrt{h}$ whose probability distribution is given by

$$P(U_n=\pm\sqrt{3})=1/6,\quad P(U_n=0)=\tfrac{2}{3}.$$

Since the density function is discrete, the integral is easily calculated to derive

$$\begin{aligned}A^6(h;\lambda,\mu)&=(1+\lambda h+\mu\sqrt{3h})(1+\lambda h)^4(1+\lambda h-\mu\sqrt{3h})\\&=(1+\lambda h)^4\{(1+\lambda h)^2-3\mu^2h\}.\end{aligned}$$

Similar to the Euler–Maruyama case, we may introduce the T-stability function for other schemes (see [16,17]).

In [5], a more practical restriction of T-stability is introduced. To avoid stability violation due to T-stability function close to 1, for a certain positive constant A less than 1 the scheme is said to be $T(A)$-stable if the T-stability function is smaller than A.

Stability analysis for numerical schemes of SDEs is still in a premature stage, although much work has been devoted to it. One of the present difficulties is, contrary to the ODE case, linear stability on the supermartingale equation cannot straightforwardly be extended to the multi-dimensional case, for then we have two matrices for the drift and the diffusion terms, not necessarily commuting with each other. Therefore, much more study is expected.

3. Fixed stepsize implementation

The first method for solving SDEs numerically was the Euler–Maruyama method which is inefficient due to its strong order of convergence $\frac{1}{2}$. Because of this limitation in order, numerical methods of higher order have been developed. Burrage and Burrage [1] have focussed their attention on stochastic Runge–Kutta methods (SRKs) of the form (for $i=1,\ldots,s$)

$$\begin{aligned}Y_i&=y_n+h\sum_{j=1}^{i-1}a_{ij}f(Y_j)+\sum_{j=1}^{i-1}\left(J_1b_{ij}^{(1)}+\frac{J_{10}}{h}b_{ij}^{(2)}\right)g(Y_j),\\y_{n+1}&=y_n+h\sum_{j=1}^{s}\alpha_jf(Y_j)+\sum_{j=1}^{s}\left(J_1\gamma_j^{(1)}+\frac{J_{10}}{h}\gamma_j^{(2)}\right)g(Y_j).\end{aligned}\tag{9}$$

If the method does not include J_{10}, then the maximum strong order is 1.0; the inclusion of this Stratonovich integral allows methods with strong order greater than 1 to be developed (see [1]). Methods formulated from (9) can be extended for use in the d-Wiener process case (as long as the SDE system coefficients are fully commutative — otherwise the order of the method is

reduced to 0.5) by sampling additionally from $2,\ldots,d$ random number generators. One way of overcoming this order reduction is to include commutators in the method formulation (see [3]). However, implementation costs are increased for methods with commutators, due to the expense of calculating derivatives, leading to the development of suitable methods without commutators [6].

A fixed stepsize implementation of a SRK involves sampling the random variables in the method (represented by J_1 and J_{10}, for example). The built-in random number generator that produces samples from a $N(0,1)$-distribution can be used; an alternative is to obtain samples from the uniform distribution and to use the Polar–Marsaglia technique to produce pairs of normally distributed random numbers. Thus, given a pair (g_1,g_2) of normally distributed random variables, $J_1=\sqrt{h}g_1$, $J_{10}=h^{3/2}(g_1+g_2/\sqrt{3})/2$.

With an initial value for the SDE, and with the means of sampling the necessary random variables, the numerical method can be implemented step by step to obtain a trajectory of the solution. However, fixed stepsize implementations of numerical methods have limitations when, for example, the SDE being solved is stiff in some subdomain of the integration as this forces the stepsize to be very small for the entire range of the integration. Thus it is natural to adapt the implementation technique to use a variable stepsize, and it is this approach that is discussed in the next section.

4. Variable stepsize implementation

In order to use a variable stepsize technique, it is necessary to estimate the error at each step so that a new and appropriate stepsize can be determined. This error estimation must be cheap, and in this paper the errors are estimated via the process of embedding. In this paper, a two-stage SRK of strong order 1 is embedded within a four-stage SRK of strong order 1.5, and the error at each step is determined by comparing the numerical results from each of the two methods; only two extra function evaluations are required to calculate the update value from the two-stage method, and so the error estimate is achieved with minimal overhead.

Let $\hat{y}_{n+1}$ be the numerical result obtained from the implementation of the s-stage method, and let y_{n+1} be that obtained from the higher stage method (where the methods have order $\hat{p}$ and p, respectively). Then y_{n+1} is used to advance the numerical computation on the next step, while both $\hat{y}_{n+1}$ and y_{n+1} are used to estimate the error. Here it is absolute error that is under consideration. For an m-dimensional system, let tol_i be the tolerance permitted for the ith component; then an error estimate of order $q+\frac{1}{2}$ (where $q=\min(\hat{p},p)$) is given by

$$\mathrm{error}=\sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(\frac{y_{n+1,i}-\hat{y}_{n+1,i}}{\mathrm{tol}_i}\right)^2}.$$

For the (R2,E1)-embedded pair of methods, in which $q=1$, we extend the variable stepsize strategy in [9], and decrease the optimal stepsize by a safety factor (for example, $\mathrm{fac}=0.8$) to avoid oscillatory behaviour in the stepsize, and place bounds so that the stepsize does not increase or decrease too quickly. Thus

$$h_{\mathrm{new}}=h\min(\mathrm{facmx},\max(\mathrm{facmn},\mathrm{fac}(1/\mathrm{error})^{2/3})), \tag{10}$$

where facmx and facmn are the maximal and minimal stepsize scaling factors allowed, respectively, for the problem being solved.

The embedded pair used to produce the numerical results in this paper consists of method R2 defined by

$$\begin{array}{c|cc|cc} 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3} & 0 & \frac{2}{3}J_1 & 0 \\ \hline & \frac{1}{4} & \frac{3}{4} & \frac{1}{4}J_1 & \frac{3}{4}J_1 \end{array}$$

and the four-stage method E1 given by (9) with parameters

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \frac{2}{3} & 0 & 0 & 0 \\ \frac{3}{2} & -\frac{1}{3} & 0 & 0 \\ \frac{7}{6} & 0 & 0 & 0 \end{pmatrix}, \quad B^{(1)} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \frac{2}{3} & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{6} & 0 & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 \end{pmatrix}, \quad B^{(2)} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{2}{3} & 0 & 0 & 0 \\ \frac{1}{6} & \frac{1}{2} & 0 & 0 \end{pmatrix},$$

$$\alpha^{\mathrm{T}} = (\tfrac{1}{4}, \tfrac{3}{4}, -\tfrac{3}{4}, \tfrac{3}{4}), \quad \gamma^{(1)\mathrm{T}} = (-\tfrac{1}{2}, \tfrac{3}{2}, -\tfrac{3}{4}, \tfrac{3}{4}), \quad \gamma^{(2)\mathrm{T}} = (\tfrac{3}{2}, -\tfrac{3}{2}, 0, 0).$$

Most implementations of numerical methods for solving SDEs use a fixed stepsize, and indeed convergence of the method was only demonstrated for such stepsizes. However, recently [7] have proved that a method must have strong order at least 1 to guarantee convergence to the correct solution if variable stepsizes are used. This result demonstrates that the embedded pair (R2,E1) will converge to the correct solution in a variable stepsize implementation.

It is important when using a variable stepsize implementation to remain on the correct Brownian path. The Brownian path consists of the Wiener increments sampled from the $N(0,1)$ distribution; these increments are scaled according to the stepsize currently being used, so when a stepsize is rejected, the increment must be rescaled in such a way that the integration remains on the true path. This approach ensures that the same Brownian path can be traversed if the numerical calculations are repeated with a different initial value or a different initial stepsize.

The approach in [7] was to use a Brownian tree of increments in the integration. The tree was formed by fixing a top level of increments with a nominal stepsize h_0 and then successively halving the stepsize and calculating the new increments on the subintervals so that the top level path was adhered to. These increments accounted for J_1, while any higher-order Stratonovich integrals (for example, J_{10}) could be calculated using a Lévy area (see [14,7,3]). At any stage of the integration, if the current stepsize h was rejected, the step would be retried with a stepsize of $h/2$, while if the step was successful the next step would proceed with either h or $2h$, depending on the alignment of the current position within the tree. This binary tree structure necessitates only a halving or doubling of stepsize, and in practice this can be too restrictive.

Another approach is in [15] who demonstrates that, given J_1 and J_{10} on a fixed Brownian path, then for $0 < h_1 < h$ and $h_2 = h - h_1$,

$$\begin{pmatrix} J_1(t_0, t_0 + h_1) \\ J_{10}(t_0, t_0 + h_1) \\ J_1(t_0 + h_1, t_0 + h) \\ J_{10}(t_0 + h_1, t_0 + h) \end{pmatrix} = AU \begin{pmatrix} N_1 \\ N_2 \\ A_h^{-1} \begin{pmatrix} j_1 \\ j_{10} \end{pmatrix} \end{pmatrix}, \tag{11}$$

is also on the same Brownian path; here $(j_1, j_{10})^{\mathrm{T}}$ are the sampled values corresponding to $J_1(t_0, t_0+h) = \int_{t_0}^{t_0+h} \circ\, \mathrm{d}W_s^{(1)}$ and $J_{10}(t_0, t_0+h) = \int_{t_0}^{t_0+h} \int_{t_0}^{s} \circ\, \mathrm{d}W_{s_1}^{(1)}\, \mathrm{d}s$ respectively, $N_1, N_2 \sim N(0,1)$,

$$A_h = \begin{pmatrix} \sqrt{h} & 0 \\ \frac{1}{2}h^{3/2} & 1/2\sqrt{3}h^{3/2} \end{pmatrix}, \quad A = \begin{pmatrix} A_{h_1} & 0 \\ 0 & A_{h_2} \end{pmatrix}, \quad \theta = \frac{h_2}{h_1},$$

$$U = \begin{pmatrix} 0 & -(\theta^2 - \theta + 1)/c_2 & 1/c_3 & \sqrt{3}\theta/c_4 \\ \theta^{3/2}/c_1 & \sqrt{3}/c_2 & 0 & 1/c_4 \\ 0 & 1 - \theta + \theta^2/c_2\sqrt{\theta} & \sqrt{\theta}/c_3 & -\sqrt{3\theta}/c_4 \\ -1/c_1 & \sqrt{3}\theta^{3/2}/c_2 & 0 & \theta^{3/2}/c_4 \end{pmatrix},$$

$$c_1 = \sqrt{\theta^3 + 1}, \quad c_2 = \sqrt{\frac{(1 - \theta + \theta^2)(1+\theta)^3}{\theta}}, \quad c_3 = \sqrt{\theta + 1}, \quad c_4 = \sqrt{(\theta+1)^3}.$$

Setting $h_1 = h_2 = h/2$ yields the transformation required when a simple halving of h takes place. Indeed, Mauthner [15] only develops this latter case, due to the ease of storing the simulated values in a binary tree as well as the reduced cost associated with their simulation.

However, in this paper, the case with arbitrary stepsize change is developed as this provides the most flexibility for a variable stepsize implementation. First, the Brownian path is fixed for a nominated stepsize h_{fix} — this can represent a series of output points, for example. If this stepsize is the maximum allowed for the integration, then all subsequent simulations are generated 'downwards'; however, if the integration requires $h > h_{\mathrm{fix}}$, the simulated Stratonovich integrals can just as easily be generated 'upwards' from the fixed path. Given the fixed Brownian path, the integration proceeds, using the desired stepsize h_1; the values of J_1 and J_{10} on these subintervals do not need to be stored — they can be merely generated as required based on the fixed path. At the end of the integration, the sum of the J_1-values along the path actually followed equals the sum of the J_1-values along the fixed path. Similarly, the J_{10}-values adhere to the definition

$$\begin{aligned} J_{10}(t_1, t_3) &= \int_{t_1}^{t_3} \int_{t_1}^{s} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s = \int_{t_1}^{t_2} \int_{t_1}^{s} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s + \int_{t_2}^{t_3} \int_{t_1}^{s} \circ\, \mathrm{d}W_{s_1}\, \mathrm{d}s, \\ &= J_{10}(t_1, t_2) + \int_{t_2}^{t_3} \left(\int_{t_1}^{t_2} \circ\, \mathrm{d}W_{s_1} + \int_{t_2}^{s} \circ\, \mathrm{d}W_{s_1} \right) \mathrm{d}s, \\ &= J_{10}(t_1, t_2) + J_{10}(t_2, t_3) + (t_3 - t_2) J_1(t_1, t_2), \end{aligned}$$

for the subintervals $[t_1, t_3] = [t_1, t_2] \cup [t_2, t_3]$. Further details using this approach, together with other examples, can be found in [4].

This section will conclude with the presentation of an example which demonstrates the efficacy of the variable stepsize approach.

Example 9. This SDE is taken from [12], (Eq. 4.4.46) and has been converted to Stratonovich form

$$\mathrm{d}y = -\alpha(1 - y^2)\, \mathrm{d}t + \beta(1 - y^2) \circ \mathrm{d}W$$

with $\alpha = 1$ and $\beta =$ each of 0.8, 1.5 and 2.0. The fixed stepsize results (using method E1) are presented in Table 1, with the variable implementation results for a range of tolerances in Table 2

Table 1
Fixed stepsize

$\beta = 0.8$			$\beta = 1.5$			$\beta = 2.0$		
h	Error	Steps	h	Error	Steps	h	Error	Steps
$\frac{1}{3}$	—	30	$\frac{1}{4}$	—	40	$\frac{1}{5}$	—	50
$\frac{1}{5}$	8.88(−5)	50	$\frac{1}{8}$	—	80	$\frac{1}{10}$	—	100
$\frac{1}{10}$	6.67(−5)	100	$\frac{1}{24}$	2.51(−2)	240	$\frac{1}{35}$	3.94(−2)	350
$\frac{1}{36}$	1.01(−5)	360	$\frac{1}{84}$	6.90(−3)	840	$\frac{1}{140}$	8.30(−3)	1400

Table 2
Variable stepsize

	$\beta = 0.8$			$\beta = 1.5$			$\beta = 2.0$		
Tol	Error	Tried	OK	Error	Tried	OK	Error	Tried	OK
0.1	1.63(−2)	29	23	2.17(−2)	44	35	2.52(−2)	105	78
0.01	1.86(−3)	47	37	1.89(−3)	81	61	3.08(−3)	112	82
0.001	1.58(−4)	114	86	1.78(−4)	261	191	2.69(−4)	399	289
0.0001	1.64(−5)	383	279	1.61(−5)	961	694	3.00(−5)	1588	1143

(average steps tried and steps accepted are given too). The initial value is 0, the integration is carried out from 0 to 10, and for the variable implementation an arbitrary initial stepsize of $\frac{1}{32}$ was used. The results were averaged over 100 trajectories.

5. Conclusions

It is clear from the discussion in this paper that stability is a critical aspect in designing useful numerical methods. Just as crucial, and what has been given less attention until recently, is that any effective implementation must consider a number of important issues, one of which is a variable stepsize implementation (under the proviso that different numerical simulations must follow the same Brownian path).

The numerical results in this paper have demonstrated that the variable stepsize implementation is far superior to that of fixed stepsize unless the stochasticity is small enough (e.g., $\beta = 0.8$) for the numerical solution to be smooth (in which case any variable stepsize implementation does not have a chance to perform under conditions suited to it). Also, although there is not exact tolerance proportionality when the tolerance is reduced by a factor of 10, the decrease in error is nearly in proportion. Clearly, our approach and the approach in [4] is very promising and offers great flexibility.

References

[1] K. Burrage, P.M. Burrage, High strong order explicit Runge–Kutta methods for stochastic ordinary differential equations, Appl. Numer. Math. 22 (1996) 81–101.
[2] K. Burrage, P.M. Burrage, General order conditions for stochastic Runge–Kutta methods for both commuting and non-commuting stochastic differential equation systems, Appl. Numer. Math. 28 (1998) 161–177.
[3] K. Burrage, P.M. Burrage, High strong order methods for non-commutative stochastic ordinary differential equation systems and the Magnus formula, in: S. Chen, L. Margolin, D. Sharp (Eds.), Predictability: Quantifying Uncertainty in Models of Complex Phenomena, Elsevier Science B.V., The Netherlands, Physica D 133(1–4) (1999) 34–48.
[4] K. Burrage, P.M. Burrage, A variable stepsize implementation for stochastic differential equations (2000), submitted for publication.
[5] K. Burrage, T.-H. Tian, The composite Euler method for stiff stochastic differential equations (2000), submitted for publication.
[6] K. Burrage, P.M. Burrage, Without-commutator methods for stochastic ordinary differential equations, in preparation.
[7] J. G. Gaines, T.J. Lyons, Variable step size control in the numerical solution of stochastic differential equations, SIAM J. App. Math 57 (5) (1997) 1455–1484.
[8] T. C. Gard, Introduction to Stochastic Differential Equations, Marcel Dekker, New York, 1988.
[9] E. Hairer, S.P. Nørsett, G. Wanner, Solving Ordinary Differential Equations I, Nonstiff Problems, 2nd Edition (revised), Springer, Berlin, 1993.
[10] R.Z. Has'minskii, Stochastic Stability of Differential Equations, Stijthoff & Noordhoff, Alphen a/d Rijn, 1980.
[11] D. J. Higham, Mean-square and asymptotic stability of numerical methods for stochastic ordinary differential equations, University of Strathclyde Mathematics Research Report, Vol. 39, 1998.
[12] P.E. Kloeden, E. Platen, Numerical Solution of Stochastic Differential Equations, Springer, Berlin, 1992.
[13] Y. Komori, T. Mitsui, H. Sugiura, Rooted tree analysis of the order conditions of ROW-type scheme for stochastic differential equations, BIT 37 (1) (1997) 43–66.
[14] P. Levy, Processus Stochastiques et Mouvement Brownien, Monographies des Probabilités, Gauthier-Villars, Paris, 1948.
[15] S. Mauthner, Thesis, Chapter 4, private communication, 1999.
[16] T. Mitsui, Stability analysis of numerical solution of stochastic differential equations, Kokyuroku (Res. Inst. Math. Sci., Kyoto Univ.) 850 (1995) 124–138.
[17] Y. Saito, T. Mitsui, T-stability of numerical scheme for stochastic differential equations, in: R.P. Agarwal (Ed.), Contributions in Numerical Mathematics, World Scientific Series in Applicable Analysis, Vol. 2, World Scientific, Singapore, 1993, pp. 333–344.
[18] Y. Saito, T. Mitsui, Stability analysis of numerical schemes for stochastic differential equations, SIAM J. Numer. Anal. 33 (1996) 2254–2267.